

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное автономное
образовательное учреждение высшего образования
«Национальный исследовательский университет ИТМО»

ФАКУЛЬТЕТ ПРОГРАММНОЙ ИНЖЕНЕРИИ И КОМПЬЮТЕРНОЙ ТЕХНИКИ

ЛАБОРАТОРНАЯ РАБОТА №1

по дисциплине

‘Распределённые системы хранения данных’

Вариант №536

Выполнил:

Студент группы Р33312

Соболев Иван

Александрович

Преподаватель:

Николаев Владимир

Вячеславович



УНИВЕРСИТЕТ ИТМО

Санкт-Петербург, 2024

Задание:

Введите вариант: 536

Используя сведения из системных каталогов, получить информацию обо всех таблицах, на которые текущий пользователь может выдать права доступа другому заданному пользователю. Полученную информацию представить в виде списка следующего формата:

```
Текущий пользователь: s100000
Кому выдаём права доступа: s111111

No.  Имя таблицы
-----
1  TABLE1
2  TABLE2
3  TABLE3
...
```

Программу оформить в виде анонимного блока.

Выполнение:

Проанализировав доступные системные каталоги, я пришел к выводу, что необходимая информация хранится в каталогах - information_schema.role_table_grants, pg_tables.

Код анонимного блока:

```
DO $$
    DECLARE
        row        record;
        from_user text := 's1000000';
        to_user text := 's1111111';
        table_num int := 1;
        name_of_table text;
    BEGIN
        RAISE INFO 'Текущий пользователь: %', from_user;
        RAISE INFO 'Кому выдаём права доступа: %', to_user;
        RAISE INFO 'No.  Имя таблицы';
        RAISE INFO '---  -----';

        FOR row IN
            SELECT DISTINCT table_name as name_of_table
            FROM information_schema.role_table_grants rg
            JOIN pg_tables pt ON rg.table_schema = pt.schemaname AND
rg.table_name = pt.tablename
            WHERE grantor = from_user and is_grantable = 'YES'
        LOOP
            RAISE INFO '% %', table_num, row.name_of_table;
            table_num := table_num + 1;
        END LOOP;
    END $$;
```

Вывод:

```
INFO: Текущий пользователь: s100000
INFO: Кому выдаём права доступа: s111111
INFO: No.  Имя таблицы
INFO: ---  -----
```

Имеем пустой вывод так, как пользователя s100000 в нашей базе нет:

mca		{
nasty		{
nikolaevv		{
ninomt		{
nnaumova		{
ozhiganov		{
pasha		{
platunov		{
polyakov		{
postgres	Суперпользователь, Создаёт роли, Создаёт БД, Репликация, Пропускать RLS	{
pvalakshin		{
roggired		{
s119391		{
s125520		{
s126685		{
s134051		{
s134057		{
s141815		{
s141821		{
s146963		{
s148787		{
s154966		{

Попробуем поменять его на доступного s119391:

```
INFO: Текущий пользователь: s119391
INFO: Кому выдаём права доступа: s111111
INFO: No. Имя таблицы
INFO: ---
```

Также имеем пустой вывод. Это связано с тем, что к базе я подключаюсь со своим пользователем, который не имеет права на просмотр таблиц и данных других пользователей. Попробуем заменить на мой номер s336760 и убедимся в работоспособности программы:

```
INFO: Текущий пользователь: s336760
INFO: Кому выдаём права доступа: s111111
INFO: No. Имя таблицы
INFO: ---
INFO: 1 administrator
INFO: 2 callcenterworker
INFO: 3 car
INFO: 4 detail
INFO: 5 detailprovider
INFO: 6 detailprovider_detail
INFO: 7 employee
INFO: 8 employee_sto
INFO: 9 images
INFO: 10 manufacturer
INFO: 11 offer
INFO: 12 offer_detail
INFO: 13 offer_purchase
INFO: 14 offer_tool
INFO: 15 purchase
INFO: 16 question
INFO: 17 review
INFO: 18 review_offer
INFO: 19 sto
INFO: 20 swaphub_images
INFO: 21 swaphub_products
INFO: 22 swaphub_user_role
INFO: 23 swaphub_users
INFO: 24 tool
INFO: 25 user_car
INFO: 26 user_role
INFO: 27 users
INFO: 28 users_questions
INFO: 29 worker
INFO: 30 worker_worker_requests
INFO: 31 workerrequest
```

Однако менять пользователей внутри кода неудобно, автоматизируем:

Создадим процедуру, в которую можно будет передавать параметры из командной строки:

```
create or replace procedure get_tables(from_user text, to_user text)
as $$
DECLARE
    row    record;
    table_num int := 1;
    name_of_table text;
BEGIN
    RAISE INFO 'Текущий пользователь: %', from_user;
    RAISE INFO 'Кому выдаём права доступа: %', to_user;
    RAISE INFO 'No.  Имя таблицы';
    RAISE INFO '---  -----';

    FOR row IN
        SELECT DISTINCT table_name as name_of_table
        FROM information_schema.role_table_grants rg
        JOIN pg_tables pt ON rg.table_schema = pt.schemaname AND
rg.table_name = pt.tablename
        WHERE grantor = from_user and is_grantable = 'YES'
    LOOP
        RAISE INFO '% %', table_num, row.name_of_table;
        table_num := table_num + 1;
    END LOOP;
END; $$
LANGUAGE 'plpgsql';
```

И также нужен скрипт запуска этого всего:

```
\echo 'Введите текущего пользователя: '
\prompt '' current_user_name

\set curr_user '\'' :current_user_name '\''

\echo 'Введите имя пользователя, кому нужно выдать права доступа: '
\prompt 'Введите имя пользователя, кому нужно выдать права доступа: '
target_user_name

\set to_user '\'' :target_user_name '\''

CALL get_tables(:curr_user::text, :to_user::text);
```

Далее запускаем наш скрипт командой:

```
psql -h pg -d studs -f ~/script.sql 2>&1 | sed 's|.*INFO: ||g'
```

и имеем вывод:

```

[s336760@helios ~]$ psql -h pg -d studs -f ~/script.sql 2>&1 | sed 's|.*INFO: ||g'
Введите текущего пользователя:
s336760
Введите имя пользователя, кому нужно выдать права доступа:
s335667
Текущий пользователь: s336760
Кому выдаём права доступа: s335667
No.  Имя таблицы
---  -----
1 administrator
2 callcenterworker
3 car
4 detail
5 detailprovider
6 detailprovider_detail
7 employee
8 employee_sto
9 images
10 manufacturer
11 offer
12 offer_detail
13 offer_purchase
14 offer_tool
15 purchase
16 question
17 review
18 review_offer
19 sto
20 swaphub_images
21 swaphub_products
22 swaphub_user_role
23 swaphub_users
24 tool
25 user_car
26 user_role
27 users
28 users_questions
29 worker
30 worker_worker_requests
31 workerrequest

```

Выводы по работе:

В ходе выполнения данной лабораторной работы я познакомился со внутренним устройством PostgreSQL, повторил `plpgsql`. Познакомился с анонимными блоками и на практике с помощью АБ узнал нужную мне информацию о таблицах.